

System Interface Services (SIS)

Technical Interchange Meeting
12 March 2013



"Developing Tomorrow's Leaders Today"



"Victory Through Training Overmatch"

Agenda

- Meeting Goals
- System Interface Services (SIS) Vision
- SIS Service Standards
 - SOAP vs REST Lessons Learned
 - Service Integration Strategy
 - Security Considerations
 - Interface Development Standards
 - Testing Web Services
- Case Studies
 - iESC
 - GuardU
- Future SIS Services, Plans, Concepts & Ideas
- Wrap-Up
- TBOC Interests

Meeting Goals

- To share standards for exchanging data among systems
- To identify the common practices (capitalizing on industry standards) that should be adopted by participating members
- Identify data needs of members and the authoritative data sources
- Identify commonalities between groups

System Interface Services (SIS) Vision

- The goal of the SIS Architecture is to integrate, synchronize, and facilitate effective and efficient use of training information systems throughout the ARMY. SIS consists of Enterprise Web services that facilitate the electronic exchange of data among these systems in support of this goal.

SIS Service Standards



"Developing Tomorrow's Leaders Today"



"Victory Through Training Overmatch"

SOAP vs REST

SOAP Service Standards

- [Web Services Description Language \(WSDL\) version 1.1](#)
- [Simple Object Access Protocol \(SOAP\) version 1.1](#)
- [WS-Interoperability \(WS-I\) Basic Profile version 1.1](#)
- [WS-Addressing version 1.0](#)
- [WS-BaseFaults version 1.2](#)
- [WS-Security X.509 Token Profile version 1.0](#)

Web Service Interoperability Technology (WSIT)

- [WS-MetadataExchange](#)
- [WS-ReliableMessaging](#)
- [WS-ReliableMessaging Policy](#)
- [WS-Security](#)
- [WS-SecurityPolicy](#)
- [WS-Trust](#)
- [WS-SecureConversation](#)
- [WS-Policy](#)
- [WS-PolicyAttachment](#)

REST – an Architectural Style

- [HTTP 1.1 protocol specification](#)
 - [HTTP Basic Authentication](#)
- [Web Application Description Language \(WADL\)](#)

SOAP vs REST

SOAP

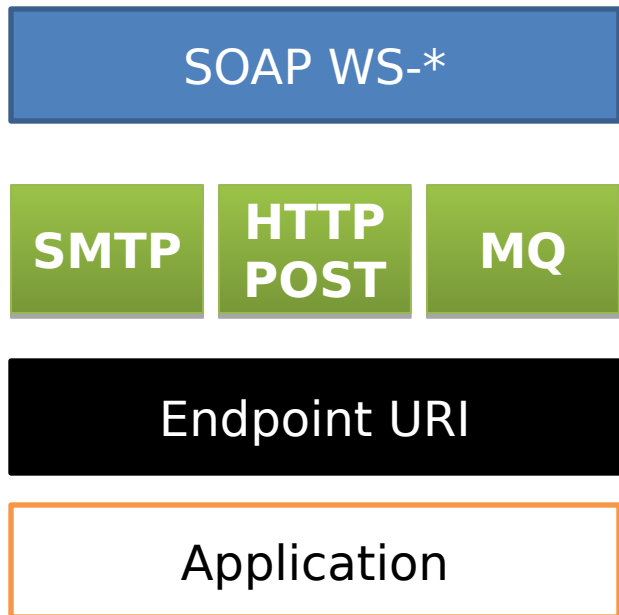
- IS a Protocol
 - HTTP POST only
 - Query/Change over POST
- dangerous not Cacheable
- Exposes Representations of Logic
- One URL
- Nothing new to discover
- XML encoded with attachments
- Bloated payload
- Machine Readable
- Toolkit needed to build
- Abstracts the network
- Change breaks clients

REST

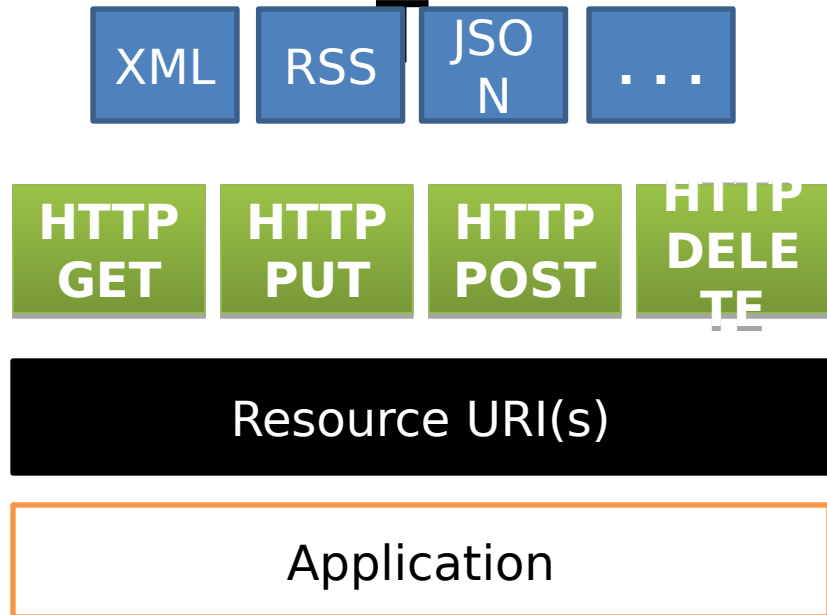
- Uses HTTP Protocol
 - GET, PUT, POST, DELETE
 - GET safe and Cacheable
- Exposes Representations of Resources
- Each URL represents a resource
- Linking resources promote discovery of new resources
- Multiple data formats
- Lightweight payload
- Human Readable
- Easy to build
- Works with the network
- Change Flexible for clients keep working

SOAP vs REST

SOAP



REST



SOAP vs REST

	SOAP /WS-*	REST
Invoke Operations	SOAP	HTTP
Transport Protocol	HTTP, TCP, SMTP, MQ ...	HTTP
Service Description	WSDL	WADL
Send Security Tokens	WS-Security	HTTP/SSL
Receive Security Tokens	WS-Trust	No Standard
Security Context	WS-SecureConversation	SSL
End-to-End Reliability	WS-ReliableMessaging	No Standard
Distributed Transactions	WS-AtomicTransaction WS-Coordination	No Standard
Defining policy	WS-Policy	No Standard
Metadata	WS-	No Standard



SOAP vs REST - Lessons Learned

- SOAP – Steep Developer Learning Curve
- SOAP – Complex WS* Security
- SOAP – High Service Creation Startup Cost
- SOAP – Difficult to Extend
- REST – Simple in Any Language plain old HTTP
- REST – Simplified Security
- REST – Quick Service Creation
- REST – Easy to Extend

Service Integration Strategy

- Service API Providers
 - Owners of Authoritative data will provide RESTful Services
 - Goal
 - Generic Design
 - Consumer Agnostic
 - Performance
 - Scalability
 - Simplicity
 - Modifiability
 - Evolvability
 - Extensibility
 - Configurability
 - Reusability
 - Discoverability via Linking

Service Integration Strategy

- Service Consumers
 - Will obtain data via RESTful Services
 - Language agnostic
 - .NET
 - Java
 - Ruby
 - PHP
 - HTML /HTML5
 - JavaScript/Ajax
 - Interface Requirement
 - HTTP 1.1

Security Considerations

- Transport Level Security
 - SSL Required
- Two Types of Authentication
 - B2B
 - HTTP Basic Authentication
 - B2C
 - Army Knowledge Online Single Sign-On (AKO SSO)
- Authorization
 - Service providers responsible for implementing authorization scheme for client access
 - Minimum of Role Based Security

Interface Development Standards

Uniform Interface

- The uniform interface is the constraint that defines the interface between the clients and servers. It simplifies and decouples the architecture which enables each part to evolve independent of one another.
- Three Elements of the REST Uniform Interface
 - Resource Identifiers
 - HTTP Methods
 - GET, POST, PUT, DELETE, HEAD, OPTIONS
 - Media Types

Interface Development Standards

Uniform Interface Continued

- Resource Identifiers
 - A resource is any item, object, idea, or thing that is important enough to be referenced in of it-self.
 - Every resource should have a unique identifier.
 - Every resource should be Addressable.
 - Use sensible names
 - Use Plural Noun for Collections
 - Eg. /api/courses
 - Use Singular Noun for Single a resource
 - Eg. /api/courses/{courseId}
 - Plural vs Singular
 - No Right Way/Wrong Way
 - BE CONSISTENT

Interface Development Standards

Uniform Interface Continued

- HTTP Methods
 - Predefined in the HTTP Specification for how to process a resource
 - GET – used to retrieve, read, query a resource
 - POST – used to create a new resource
 - PUT – used to update an existing resource
 - DELETE – used to delete an existing resource
 - HEAD – identical to GET without resource, retrieve metadata
 - OPTIONS – used to retrieve communications information
 - Manipulation of Resources through Representations
 - Don't tunnel POST/PUT/DELETE actions through GET

Interface Development Standards

Uniform Interface Continued

- Media Types
 - Identifies what type of data is being transferred
 - Representations of Resources
 - Expressed in HTTP HEADERS
 - Consumer Eg. Accept: application/xml
 - Service Eg. Content-Type: application/json
 - Optional Query parameter: api/resources?format=xml
- Representations
 - Provide at least JSON/JS and XML
 - XML – typically for B2B
 - JSON/JS – typically for B2C

Interface Development Standards

Additional Considerations

- Linking Resources
 - Promotes Self-Describing Messaging
 - Where to go next
 - Pagination
 - Late binding – ID's are discovered at runtime
 - Change does not break client
- URI Templating
- Stateless

Interface Development Standards

Additional Considerations Continued

- API Versioning
 - Custom Header X-API-VERSION: v1.0
 - Content-type: application/vnd.sis.{service}.v1+xml
 - May have client issues
- Error Handling
 - Use the HTTP Status codes properly

Interface Development Standards

Continued

- Documentation
- Documentation
- Documentation

Interface Development Standards Documentation

- Each service method should include:
 1. HTTP Method
 2. URI of each Resource
 3. Accept and Content-Type HTTP Request Headers
 4. All path, query and form parameters including, datatypes, formats, and descriptions
 5. All possible HTTP Response codes
 6. Any custom Headers
 7. All security access roles and service authentication types supported ie. SSO, B2B
 8. Sample Response
 9. Sample request body for PUT, POST requests
 10. Applicable Schema (XSDs for each request and response)

Testing Web Services

- SoapUI – Probably the most complete testing tool available for testing both SOAP and REST services. Able to set up mock services and test suites.
 - Download - <http://sourceforge.net/projects/soapui/files/>
 - Getting Started Guide
<http://www.soapui.org/REST-Testing/getting-started.html>
- Google Rest Client – RESTful web service test tool.
 - Download - <http://code.google.com/p/rest-client/>
- RESTClient - Firefox Plug-in
- Fiddler – Internet Explorer Plug-in

CASE STUDIES



"Developing Tomorrow's Leaders Today"

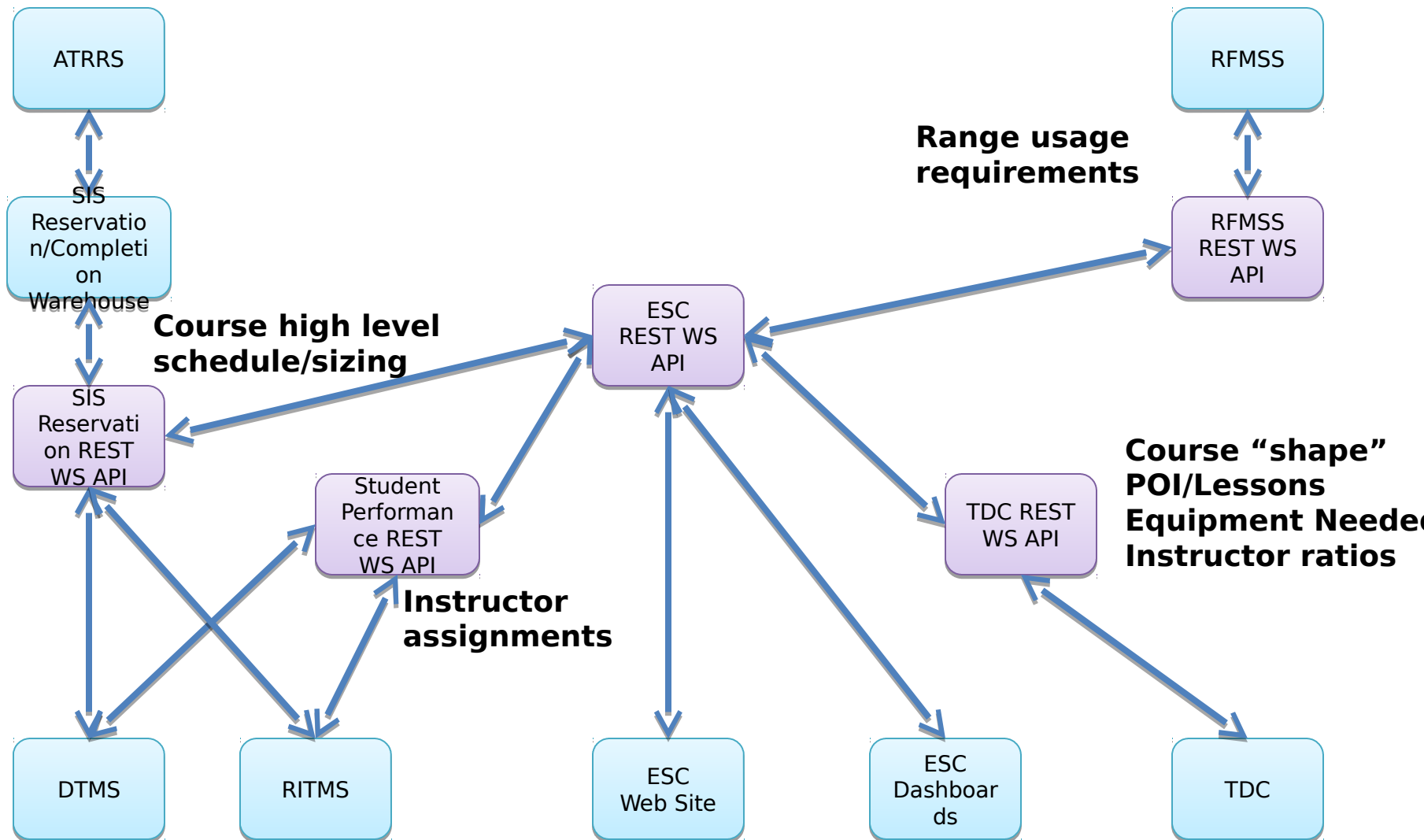


"Victory Through Training Overmatch"

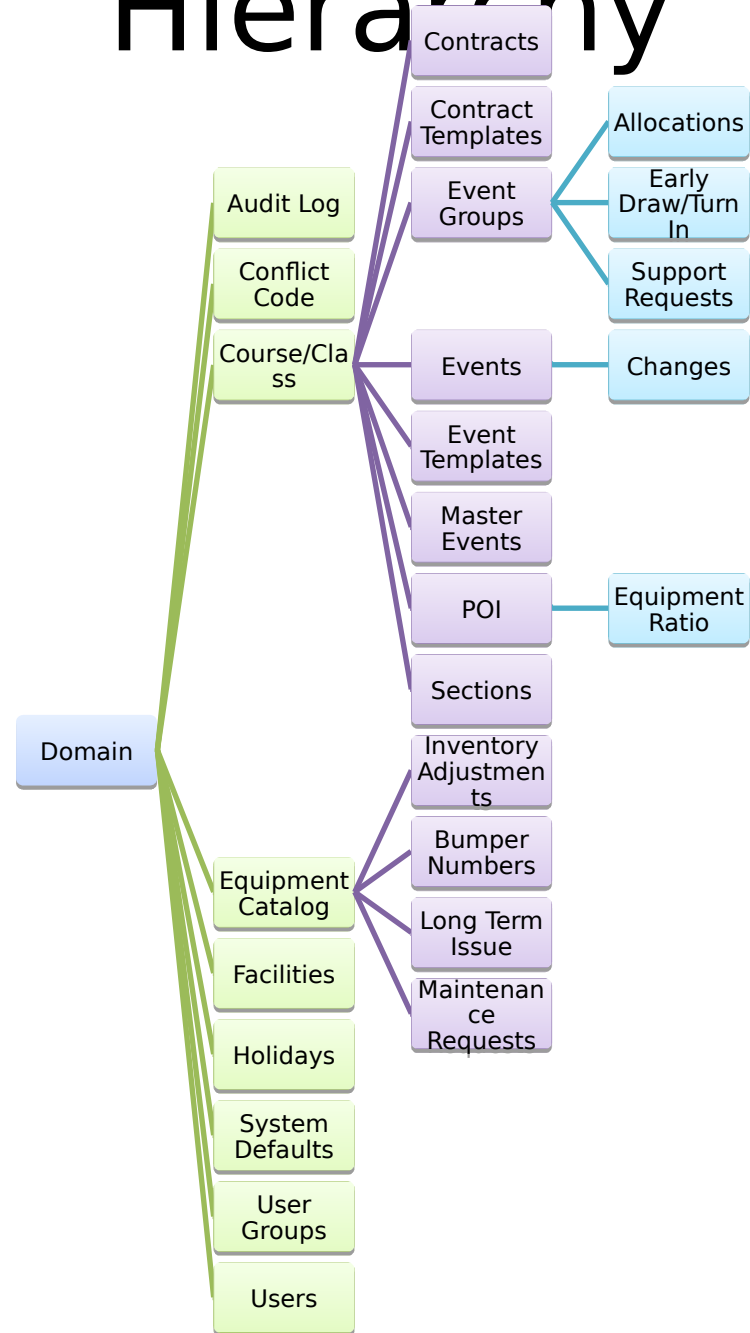
Enterprise Scheduling Capability (ESC) Implementation Strategy

- ESC will utilize the TCM-ATIS SIS architecture.
 - REST-based web service server tier
 - Use of REST and SOAP services to obtain official data sources

ESC and SIS Architecture

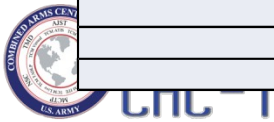


ES&RS Resource Hierarchy



Sample ESC REST WS URIs

URI Pattern	Core Resource	Service Type		HTTP Methods						Representations		
		B2C	B2B	GET	POST	PATCH	DELETE	HEAD	OPTIONS	XML	JSON	Other
\api\courseClass\{ccid}	CourseClass			X		X	X			X	X	
\api\courseClass\{ccid}\contracts	Contracts			X	X					X	X	
\api\courseClass\{ccid}\contractTemplates	Contract Templates			X	X					X	X	
\api\courseClass\{ccid}\eventGroups	Event Group			X	X					X	X	
\api\courseClass\{ccid}\events	Event			X	X					X	X	
\api\courseClass\{ccid}\eventTemplates	Master Events			X	X					X	X	
\api\courseClass\{ccid}\masters	Master			X	X					X	X	
\api\courseClass\{ccid}\poino	POI			X	X					X	X	
\api\courseClass\{ccid}\sections	Section			X	X					X	X	
\api\domain\{dmnid}	Domain			X		X				X	X	
\api\domain\{dmnid}\auditLog	Audit Log			X	X					X	X	
\api\domain\{dmnid}\auditLog\{logid}	Audit Log			X		X	X			X	X	
\api\domain\{dmnid}\confCode	Conflict Code			X	X					X	X	
\api\domain\{dmnid}\courseClasses	CourseClass			X	X					X	X	
\api\domain\{dmnid}\equipmentCatalog	Catalog			X	X					X	X	
\api\domain\{dmnid}\facilities	Facilities			X	X					X	X	
\api\domain\{dmnid}\holidays	Holidays			X	X					X	X	
\api\domain\{dmnid}\systemDefaults	SystemDefaults			X	X					X	X	
\api\domain\{dmnid}\userGroups	UserGroup			X	X					X	X	
\api\domain\{dmnid}\users	User			X	X					X	X	
\api\domains	Domain			X	X					X	X	
\api\equipment\{equipid}	Equipment			X		X	X			X	X	
\api\equipment\{equipid}\adjustments	Equipment			X	X					X	X	
\api\equipment\{equipid}\adjustments\{adjid}	Equipment			X		X	X			X	X	
\api\equipment\{equipid}\bumpers	BumperNumbers			X	X					X	X	



GuardU Case Study

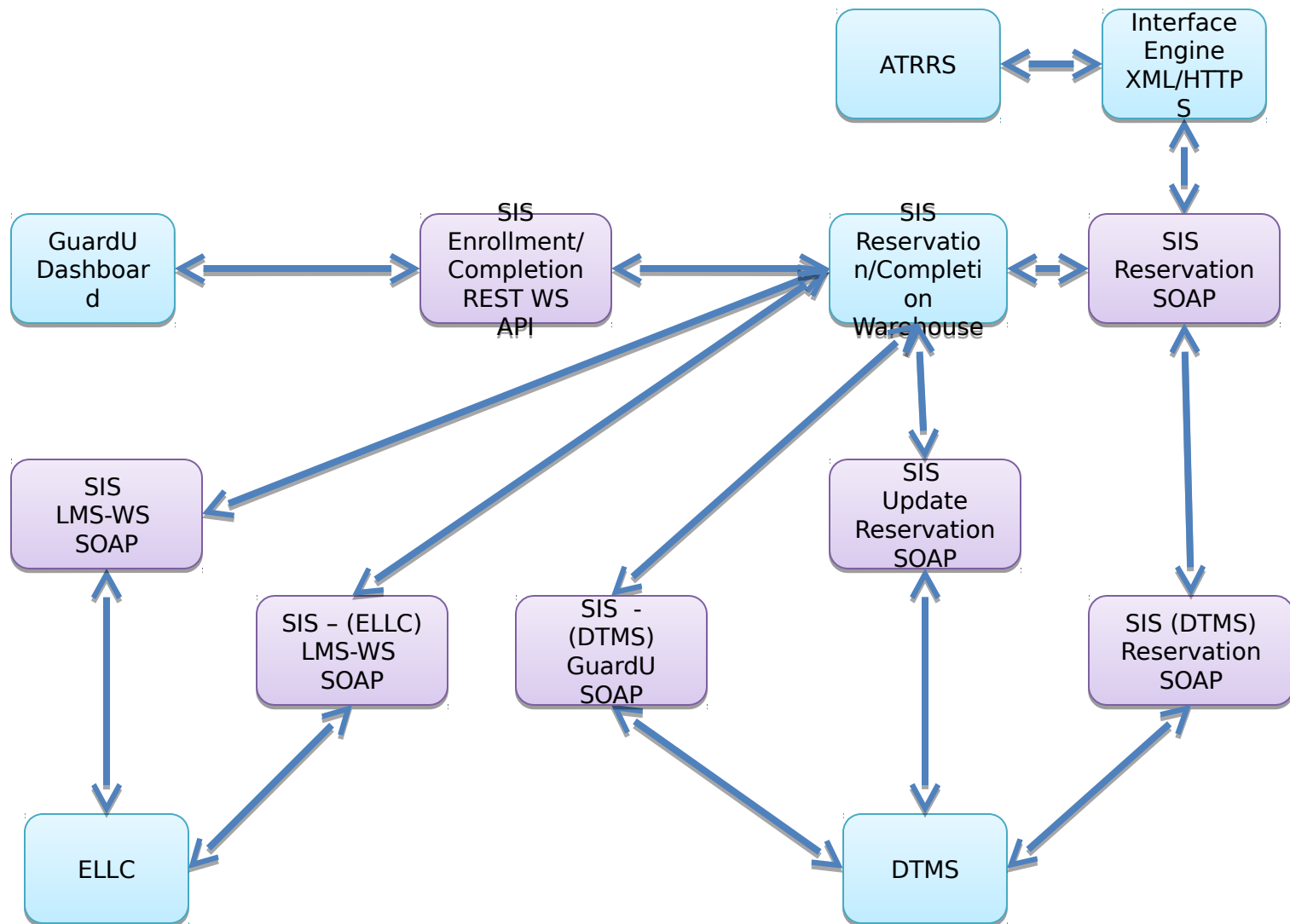
- Requirement to Report Completions by State
 - ELLC utilizes existing SIS SOAP Service
 - SIS utilizes existing DTMS SOAP Service
- GuardU Dashboard
 - Mobile First Development
 - Utilizes new RESTful Completion Service
 - GET Completions by State
 - Paginated results – Service provided links
 - Optional Filter by Date Range
 - CSV download
 - AKO SSO Enabled
 - Planned RESTful Enrollment Service

GuardU Demo

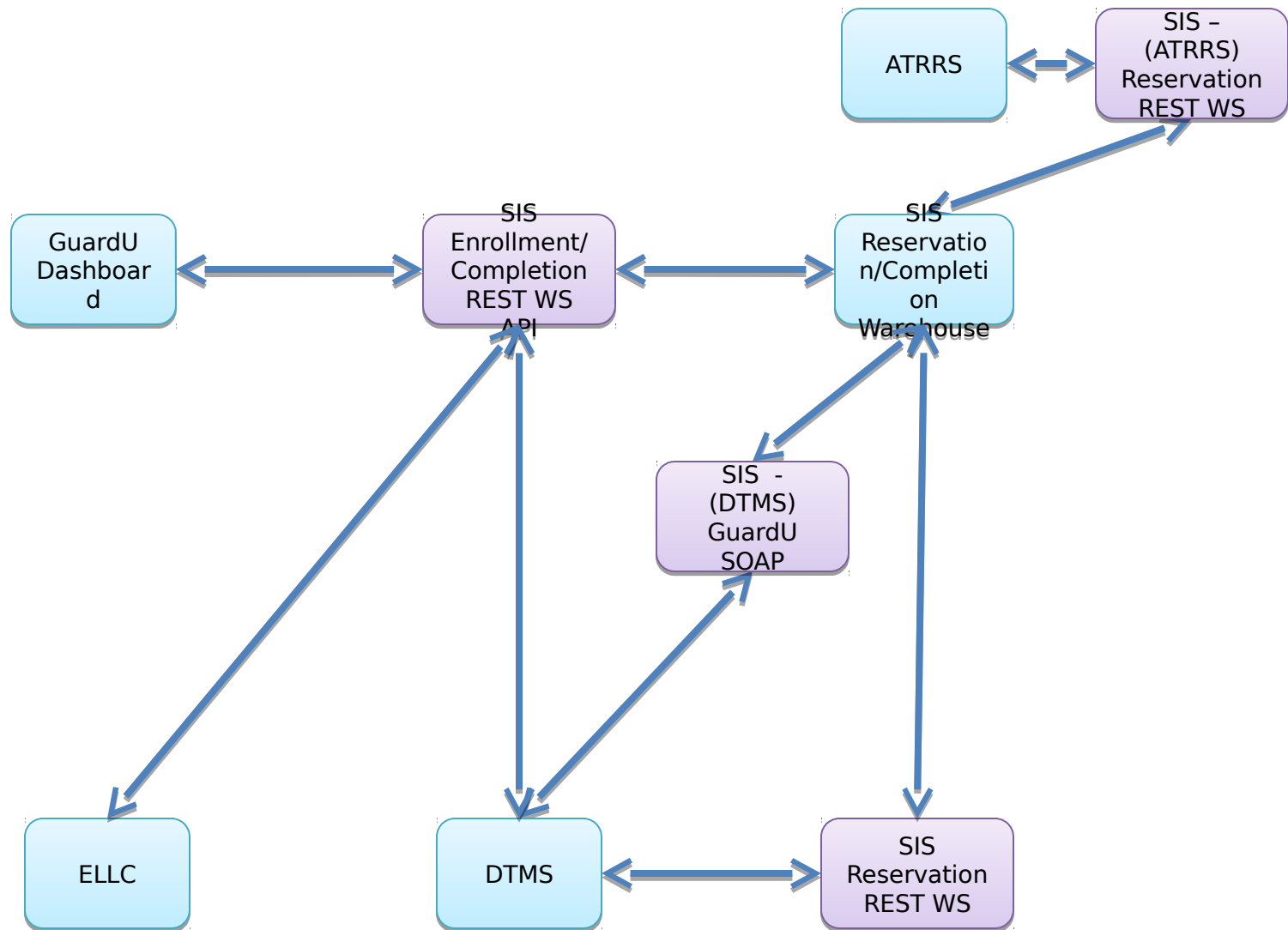
<https://atiatest.train.army.mil/mthp/>

<https://atiatest.train.army.mil/guardu>

GuardU and Existing SIS Architecture



GuardU and Projected SIS Architecture



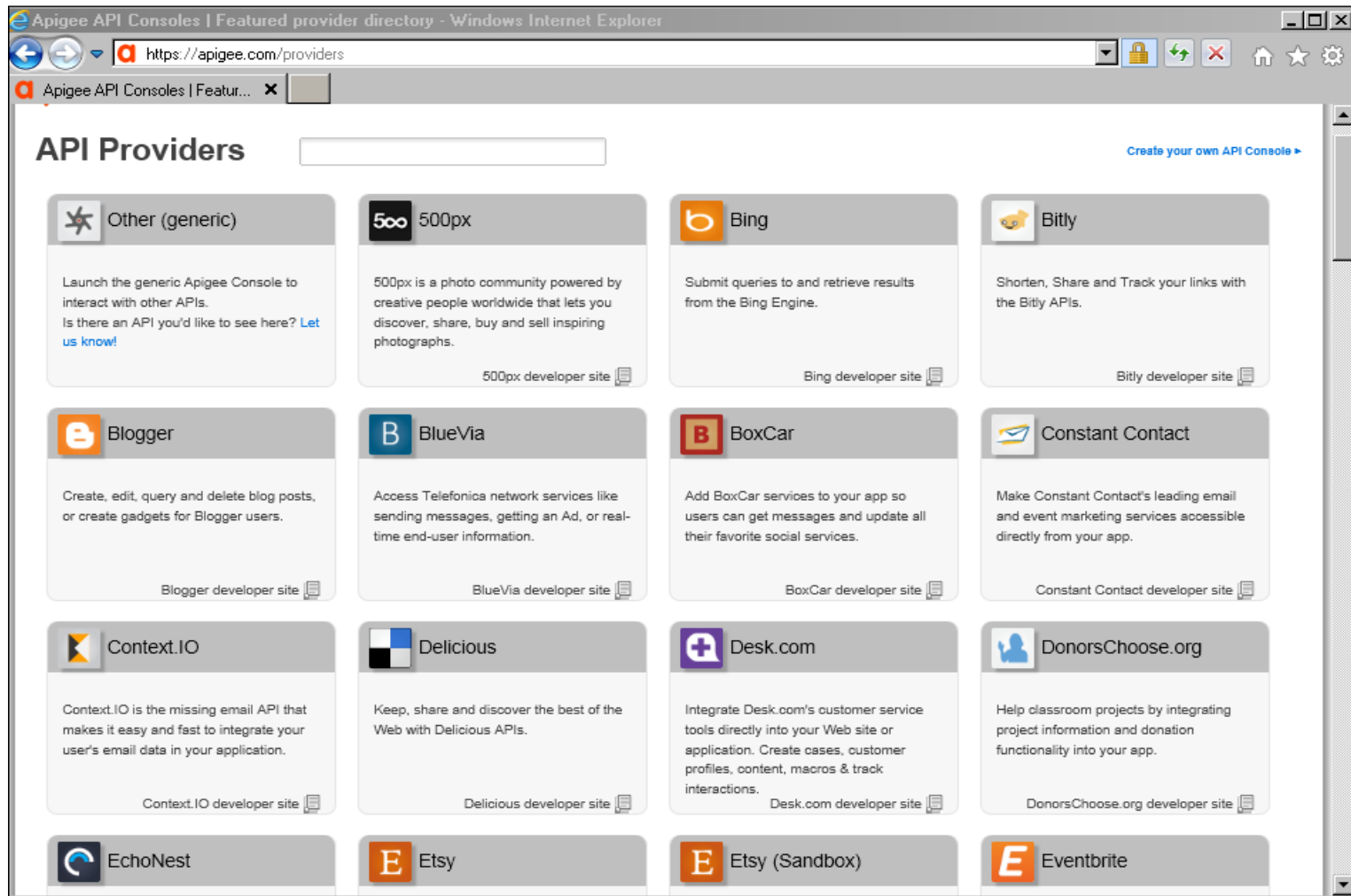
Completion Service URIs

Base URI = /sis-services/api	Status		Resource	Service Type		HTTP Methods						Representations				
Base URI +	Impl.	Planned		B2C	B2B	GET	POST	PUT	DELETE	HEAD	OPTIONS	JS	JSON	XML	CSV	Other
/application.wadl	X		WADL	X	O	X				X	X			X		
/cmpls	X		cmpls	X	O	X	X			X	X	X	X	X		
/cmpls?state=VA	X		state	X	O	X				X	X	X	X	X	X	
/cmpls?course=750-BT		O	course	O	O	X				X	X	X	X	X		
/enrls		O	enrls	O	O	O	O			O	O	O	O	O		
/enrls?state=VA		O	state	O	O	O				O	O	O	O	O	O	
/enrls?course=750-BT		O	course	O	O	O				O	O	O	O	O		

Future SIS Services, Plans, Concepts & Ideas

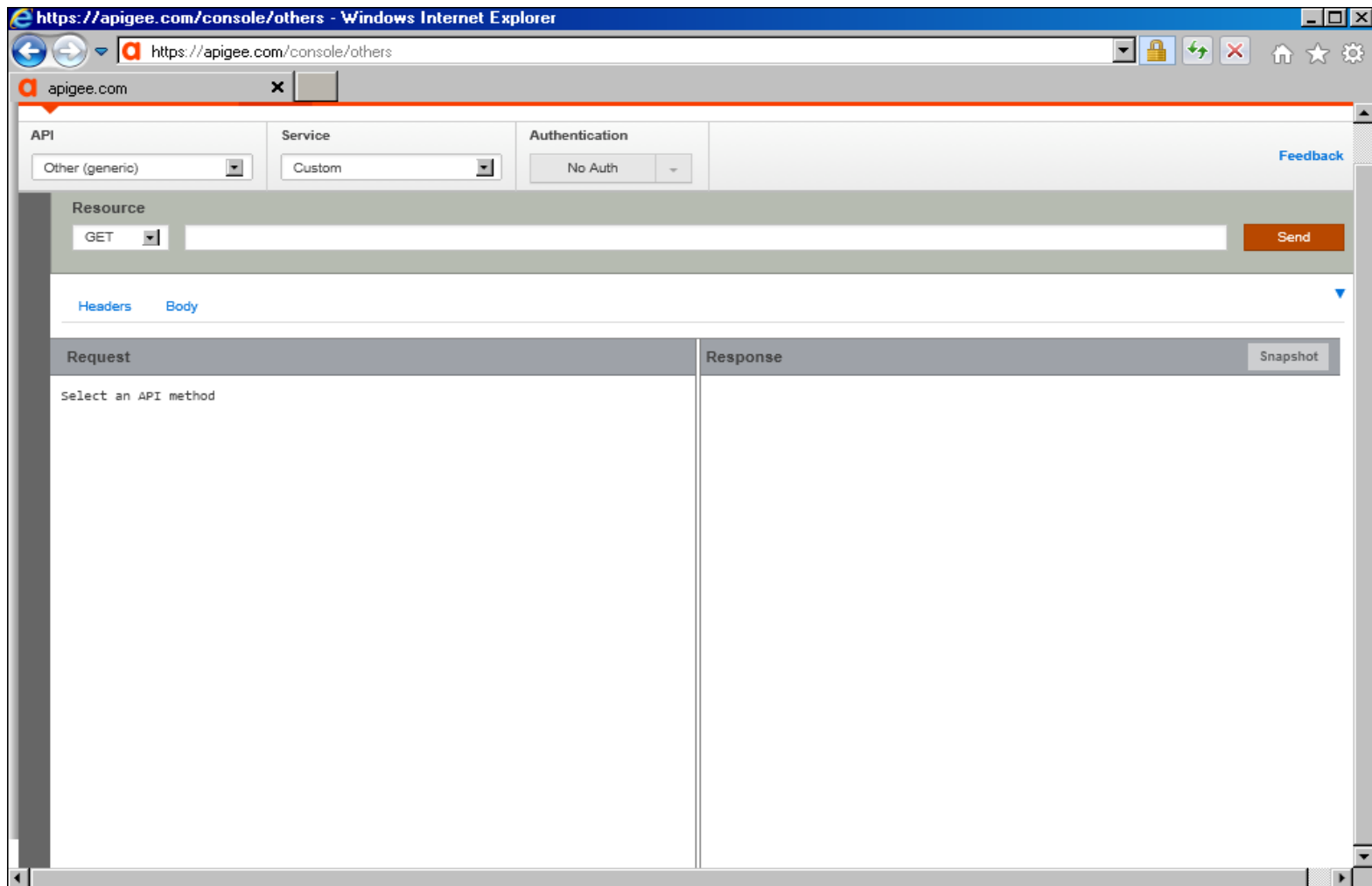
- Service Authorization API
 - Service that retrieves, creates, updates, deletes roles assigned to a user for a particular service
- Service Registry API
 - Service that retrieves, creates, updates, deletes services in the SIS portfolio.
- Service Discovery Web Site for Humans
 - Provides information to interested persons on all services available in SIS
 - Allows interested parties to request information on how to participate as a Service Provider or Consumer

API Service Provider Console Concept for SIS



CAC-T

API Service Provider Console Concept for SIS



CAC-T

TBOC Interests

- Establishment of a formal agreement in place between ATIS and TBOC to ensure Training Brain Repository (TBR) links to ATIS authoritative data sources
- TCM-ATIS and TBOC are working closely with monthly technical working group to ensure the TBR links to our web services for CATS, METL and other authoritative data.
- Assistance with NIPR CAC Log-on script and SIPR, if applicable, for the TBR

Ideas for SIS?

All ideas welcome



"Developing Tomorrow's Leaders Today"



"Victory Through Training Overmatch"

Wrap Up

- Next Steps
 - Identify who needs data
 - Who provides data
- Next Meeting